A Privacy-preserving Disaggregation Algorithm for Non-intrusive Management of Flexible Energy

Paulin Jacquot, Olivier Beaude, Pascal Benchimol, Stéphane Gaubert, Nadia Oudjane

A Privacy-preserving Disaggregation Algorithm for Non-intrusive Management of Flexible Energy

Paulin Jacquot, Olivier Beaude, Pascal Benchimol, Stéphane Gaubert, Nadia Oudjane

***Abstract*— We consider a resource allocation problem involving a large number of agents with individual constraints subject to privacy, and a central operator whose objective is to optimize a global, possibly non-convex, cost while satisfying the agents' constraints. We focus on the practical case of the management of energy consumption flexibilities by the operator of a microgrid. This paper provides a privacy-preserving algorithm that does compute the optimal allocation of resources, avoiding each agent to reveal her private information (constraints and individual solution profile) neither to the central operator nor to a third party. Our method relies on an aggregation procedure: we maintain a global allocation of resources, and gradually disaggregate this allocation to enforce the satisfaction of private constraints, by a protocol involving the generation of polyhedral cuts and secure multiparty computations (SMC). To obtain these cuts, we use an alternate projections method à la Von Neumann, which is implemented locally by each agent, preserving her privacy needs. Our theoretical and numerical results show that the method scales well as the number of agents gets large, and thus can be used to solve the allocation problem in high dimension, while addressing privacy issues.**

I. INTRODUCTION

Motivation. Consider an operator of an electricity microgrid optimizing the joint production schedules of renewable and thermal power plants in order to satisfy, at each time period, the consumption constraints of its consumers. To optimize the costs and the renewables integration, this operator relies on demand response techniques, that is, taking advantage of the flexibilities of some of the consumers electric appliances—those which can be controlled without impacting the consumer's comfort, as electric vehicles or water heaters [1]. However, for privacy reasons, consumers are not willing to provide neither their consumption constraints nor their consumption profiles to a central operator or any third party, as this information could be used to induce private information such as their presence at home.

The *global problem* of the operator is to find an allocation of power (aggregate consumption) $\boldsymbol{p} = (p_t)_t$ at each time period (*resource*) $t \in \mathcal{T}$, such that $\boldsymbol{p} \in \mathcal{P}$ (feasibility constraints of power allocation, induced by the power plants constraints). Besides, this aggregate allocation has to match an individual consumption profile $\boldsymbol{x}_n = (x_{n,t})_{t\in\mathcal{T}}$ for each of the consumer (agent) $n \in \mathcal{N}$ considered. The problem can be written as follows:

$$\min_{\boldsymbol{x}\in\mathbb{R}^{N\times T},\ \boldsymbol{p}\in\mathcal{P}} f(\boldsymbol{p}) \tag{1a}$$

$$\boldsymbol{x}_n \in \mathcal{X}_n,\ \forall n \in \mathcal{N} \tag{1b}$$

$$\textstyle\sum_{n\in\mathcal{N}} x_{n,t} = p_t,\ \forall t \in \mathcal{T}\ , \tag{1c}$$

The (aggregate) allocation $\boldsymbol{p}$ can be made *public*, that is, revealed to all agents. However, the individual constraint set $\mathcal{X}_n$ and individual profiles $\boldsymbol{x}_n$ constitute *private* information of agent n, and should not be revealed to the operator or any third party. It will be helpful to think of Problem (1) as the combination of two interdependent subproblems:

i) given an aggregate allocation $\boldsymbol{p}$, the *disaggregation problem* consists in finding, for each agent n, an individual profile $\boldsymbol{x}_n$ satisfying her individual constraint (1b), so that constraint (1c) is satisfied; when this is possible, we say that a *disaggregation* exists for $\boldsymbol{p}$;

ii) each subset $\mathcal{Q} \subset \mathcal{P}$ determines an *optimal resource allocation problem*, or *master problem*, $\min_{\boldsymbol{p}\in\mathcal{Q}} f(\boldsymbol{p})$.

When $\mathcal{Q}$ is precisely the set of aggregate allocations for which a disaggregation exists, the optimal solutions of the master problem correspond to the optimal solutions of (1).

Aside from the example above, resource allocation problems (optimizing common resources shared by multiple agents) find many applications in energy [1, 2], logistics [3], distributed computing [4], health care [5] and telecommunications [6]. In these applications, several entities or agents (e.g. consumers, stores, tasks) share a common resource (energy, products, CPU time, broadband) which has a global cost for the system. For large systems composed of multiple agents, the dimension of the overall problem can be prohibitive and one can rely on decomposition and distributed approaches [7–9] to answer to this issue. Besides, agents' individual constraints are often subject to privacy issues [10]. These considerations have paved the way to the development of privacy-preserving, or non-intrusive methods and algorithms, e.g. [11, 12].

In this work, we consider that each agent has a global demand constraint (e.g. energy demand or product quantity), which confers to the disaggregation problem the particular structure of a transportation polytope [13]: the sum over the agents is fixed by the aggregate solution $\boldsymbol{p}$, while the sum over the T resources are fixed by the agent global demand constraint. Besides, individual constraints can also include minimal and maximal levels on each resource, as for instance electricity consumers require, through their appliances, a minimal and maximal power at each time period.

P.Jacquot, O.Beaude, P.Benchimol and N.Oudjane are with EDF R&D, OSIRIS, Palaiseau, France. P.Jacquot and S.Gaubert are with Inria Saclay and CMAP, Ecole polytechnique, Palaiseau, France.

{paulin.jacquot,stephane.gaubert}@inria.fr, {olivier.beaude,pascal.benchimol,nadia.oudjane}@edf.fr

Main Results. The main contribution of the paper is to provide a non-intrusive and distributed algorithm (Algo. 4) that computes an aggregated resource allocation $\boldsymbol{p}$, optimal solution of the—possibly nonconvex—optimization problem (1), along with feasible individual profiles $\boldsymbol{x}$ for agents, without revealing the individual constraints of each agent to a third party, either another agent or a central operator. The algorithm solves iteratively instances of *master problems* $\min_{\boldsymbol{p}\in\mathcal{P}^{(s)}} f(\boldsymbol{p})$ by constructing successive approximations $\mathcal{P}^{(s)} \subset \mathcal{P}$ of the aggregate feasible set of (1) for which a disaggregation exists, by adding a new constraint on $\boldsymbol{p}$ to $\mathcal{P}^{(s)}$, before solving the next master problem.

To identify whether or not disaggregation is feasible and to add a new constraint in the latter case, our algorithm relies on the alternating projections method (APM) [14, 15] for finding a point in the intersection of convex sets. Here, we consider the two following sets: on the one hand, the affine space defined by the aggregation to a given resource profile, and on the other hand, the set defined by all agents individual constraints (demands and bounds). As the latter is defined as a Cartesian product of each agent's feasibility set, APM can operate in a distributed fashion. The sequence constructed by the APM converges to a single point if the intersection of the convex sets is nonempty, and it converges to a periodic orbit of length 2 otherwise. Our key result is the following: if the APM converges to a periodic orbit, meaning that the disaggregation is not feasible, we construct from this orbit a polyhedral *cut*, i.e. a linear inequality satisfied by all feasible solutions $\boldsymbol{p}$ of the global problem (1), but violated from the current resource allocation (Thm. 4). Adding this cut to the master problem, we can recompute a new resource allocation and repeat this procedure until disaggregation is possible. Another main result stated in this paper is the explicit upper bound on the convergence speed of APM in our framework (Thm. 2), which is obtained by spectral graph theory methods, exploiting also geometric properties of transportation polytopes. This explicit speed shows a linear impact of the number of agents, which is a strong argument for the applicability of the method in large distributed systems.

Related Work. A standard approach to solve resource allocation problems in a distributed way is to use a Lagrangian (dual) decomposition technique [8, 16, 17]. Those techniques are generally used to decompose a large problems into several subproblems of small dimension. However, Lagrangian decomposition methods are based on strong duality property, requiring global convexity hypothesis which are not satisfied in many practical problems (e.g. MILP, see Sec. V). On the contrary, our method can be used when the master allocation problem is not convex. In [2], the authors study a disaggregation problem similar to the one considered in this paper. Their results concern *zonotopic* sets, which is different from the structure we described in Sec. II. The APM has been the subject of several works in itself [15, 18, 19]. The authors of [20] provide general results on the convergence rate of APM for semi-algebraic sets. They show that the convergence is geometric for polyhedra. However, it is generally hard to compute explicitly the geometric convergence rate of APM, as this requires to bound the singular values of certain matrices arising from the polyhedral constraints. In [21], the authors provide an explicit convergence rate for APM on a class of polyhedra arising in submodular optimization. The sets they consider differ from the present transportation polytopes.

Structure. In Sec. II, we describe the master resource allocation problem and formulate the associated disaggregation problem. In Sec. III, we focus on the APM and state our main results. In Sec. IV, we apply these results to describe a non-intrusive version of APM (NI-APM) that is used to describe our non-intrusive algorithm for computing an optimal resource allocation. Finally, in Sec. V, we provide a concrete numerical example based on a MILP to model the management of a local electricity system (microgrid), and study numerically the influence of the number of agents on the time needed for convergence of our algorithm.

Notation. Vectors and matrices are denoted by bold fonts, $\boldsymbol{v}^\top$ denotes the transpose of $\boldsymbol{v}$, $\mathbb{1}_K$ denotes the vector $(1\dots1)^\top$ of size K, $\mathcal{U}([a,b])$ stands for the uniform distribution on $[a,b]$. We use $\|\boldsymbol{x}\|_2$ to denote the Frobenius norm $\|\boldsymbol{x}\|^2 = \sum_{n,t} x_{n,t}^2$, and $P_C(.)$ to denote the Euclidean projection on a convex set C.

II. Master problem and disaggregation

In this work, we suppose an operator wishes to determine an allocation of resources, represented by a T-dimensional vector $\boldsymbol{p}$, in order to minimize a global cost function f, for instance, an electricity power economic dispatch (or the allocation of different types of merchandise in warehouses in logistics applications) subject to a set of constraints described by a feasibility set $\mathcal{P}$. This problem can be nonconvex either because of nonconvex costs f or because of a nonconvex feasible set $\mathcal{P}$ (see Sec. V). In the proposed method, the operator will consider *master problems* of the form:

$$\min_{\boldsymbol{p}\in\mathcal{P}^{(s)}} f(\boldsymbol{p}) \tag{2}$$

where the set $\mathcal{P}^{(s)} \subset \mathcal{P}$ is an aggregate approximation of disaggregation constraints. Indeed, the resource allocation $\boldsymbol{p}$ has to be shared between N agents (e.g. consumers). Each agent has a global demand (total energy needed) E_n and some lower and upper bounds on each of the resource $t \in \mathcal{T}$. The admissible set of profiles of agent n is therefore:

$$\mathcal{X}_n \stackrel{\text{def}}{=} \{\boldsymbol{x}_n \in \mathbb{R}^T \mid \boldsymbol{x}_n^\top \mathbb{1}_T = E_n \text{ and } \forall t, \underline{x}_{n,t} \le x_{n,t} \le \overline{x}_{n,t}\}. \tag{3}$$

The *disaggregation problem* consists in finding individual profiles $\boldsymbol{x} = (x_{n,t})_{n,t} \in \mathbb{R}^{NT}$ of a given aggregated allocation $\boldsymbol{p}$ such that $\boldsymbol{x}_n$ is feasible for each agent n:

$$\text{FIND } \boldsymbol{x} \in \mathcal{Y}_{\boldsymbol{p}} \cap \mathcal{X} \tag{4}$$

$$\text{where } \mathcal{Y}_{\boldsymbol{p}} \stackrel{\text{def}}{=} \{\boldsymbol{y} \in \mathbb{R}^{NT} | \boldsymbol{y}^\top \mathbb{1}_N = \boldsymbol{p}\} \text{ and } \mathcal{X} \stackrel{\text{def}}{=} \prod_{n\in\mathcal{N}} \mathcal{X}_n \ .$$

Following (4), the *disaggregated* profile refers to $\boldsymbol{x}$, while the *aggregated* profile refers to the allocation $\boldsymbol{p}$.

Problem (4) may not always be feasible. Immediate necessary conditions for a solution to exist are obtained by *aggregating* the individual constraints on $\mathcal{N}$ as:

$$\boldsymbol{p}^\top \mathbb{1}_T = \boldsymbol{E}^\top \mathbb{1}_N \text{ and } \underline{\boldsymbol{x}}^\top \mathbb{1}_N \leq \boldsymbol{p} \leq \overline{\boldsymbol{x}}^\top \mathbb{1}_N \ . \tag{5}$$

However, (5) are not sufficient conditions, as shown in Fig. 1 where the problem (4) is represented as a *flow* or *circulation* problem from source nodes $t \in \mathcal{T}$ to sink nodes $n \in \mathcal{N}$.

Indeed, with this circulation representation of the disaggregation problem (4), an immediate consequence of Hoffmann theorem [22, Thm. 3.18][23] is the following characterization of the disaggregation feasibility, which involves an exponential number of inequalities:

Theorem 1. *The disaggregation problem* (4) *is feasible (i.e.* $\mathcal{X} \cap \mathcal{Y}_{\boldsymbol{p}} \neq \emptyset$*) iff for any* $\mathcal{T}_{\text{in}} \subset \mathcal{T}, \mathcal{N}_{\text{in}} \subset \mathcal{N}$*:*

$$\sum_{t \notin \mathcal{T}_{\text{in}}} p_t \leq \sum_{t \notin \mathcal{T}_{\text{in}}, n \in \mathcal{N}_{\text{in}}} \overline{x}_{n,t} - \sum_{t \in \mathcal{T}_{\text{in}}, n \notin \mathcal{N}_{\text{in}}} \underline{x}_{n,t} + \sum_{n \notin \mathcal{N}_{\text{in}}} E_n. \tag{6}$$

The inequality (6) has a simple interpretation: the residual demand (the left hand side composed of demand and exports minus production) in $\mathcal{T}_{\text{in}} \cup \mathcal{N}_{\text{in}}$ cannot exceed the import capacity (right hand side of the inequality). One can see that, in the example of Fig. 1, inequality (6) does not hold when using the cut composed of the dashed nodes p_1 and E_1.

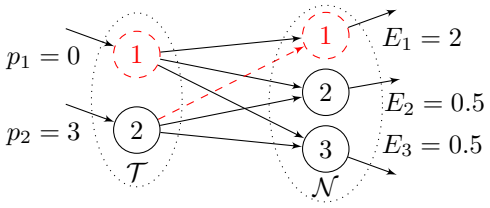

Fig. 1. Example of disaggregation structure ($T = 2, N = 3$), with $\underline{\boldsymbol{x}} = \mathbf{0}$ and $\overline{\boldsymbol{x}} := \mathbf{1}$. *Although the aggregate constraints* (5) *are satisfied, the disaggregation* (4) *of* $\boldsymbol{p}$ *is not feasible in this example (see Thm. 1).*

There are two main reasons for which solving (1) is harder than solving (II) and (4) separately:

i) the dimension of (1) can be huge, as the number of agents N can be really important, for instance in the example of individual consumers;

ii) also, and this is the main motivation of this work, the information related to $(\overline{\boldsymbol{x}}_n)_n, (\underline{\boldsymbol{x}}_n)_n$ and $(E_n)_n$ might not be available to the centralized operator in charge of optimizing resources $\boldsymbol{p}$, as this information may be confidential and kept by each agent n, not willing to reveal it to any third party.

In the next sections, we provide a method that addresses those two issues, by considering subproblems (II) and (4) independently and iteratively, and exploiting the decomposable structure of problem (4).

III. Alternate Projection Method (APM)

A. Convergence of APM on Transportation Polytopes

In this section, we consider a fixed aggregated profile $\boldsymbol{p}$ and present the Von Neumann Alternate Projections Method (APM) [14] which solves the problem Eq. (4) of finding a point in the intersection $\mathcal{X} \cap \mathcal{Y}_{\boldsymbol{p}}$. In the remaining, we will often omit $\boldsymbol{p}$ and just write $\mathcal{Y}$ to denote $\mathcal{Y}_{\boldsymbol{p}}$. The key idea of the method proposed in this paper is to use results of APM to generate a cut in the form of (6) and to add it as a new constraint in the master problem (II) to "improve" the aggregated profile $\boldsymbol{p}$ for the next iteration. As described in Algo. 1, APM can be used to decompose (4) and only involves *local* operations.

Algorithm 1 Alternate Projections Method (APM)

Require: Start with $\boldsymbol{y}^{(0)}$, $k = 0$, ε_{cvg}, a norm $\|.\|$ on $\mathbb{R}^{NT}$

1: **repeat**
2: $\quad \boldsymbol{x}^{(k+1)} \leftarrow P_{\mathcal{X}}(\boldsymbol{y}^{(k)})$
3: $\quad \boldsymbol{y}^{(k+1)} \leftarrow P_{\mathcal{Y}}(\boldsymbol{x}^{(k+1)})$
4: $\quad k \leftarrow k + 1$
5: **until** $\left\|\boldsymbol{y}^{(k)} - \boldsymbol{y}^{(k-1)}\right\| < \varepsilon_{\text{cvg}}$

The convergence of Algo. 1 is proved by Thm. 2:

Theorem 2 ([15])**.** *Let* $\mathcal{X}$ *and* $\mathcal{Y}$ *be two convex sets with* $\mathcal{X}$ *bounded, and let* $(\boldsymbol{x}^{(k)})_k$ *and* $(\boldsymbol{y}^{(k)})_k$ *be the two infinite sequences generated by Algo. 1 with* $\varepsilon_{cvg} = 0$*. Then there exists* $\boldsymbol{x}^\infty \in \mathcal{X}$ *and* $\boldsymbol{y}^\infty \in \mathcal{Y}$ *such that:*

$$\boldsymbol{x}^{(k)} \underset{k\to\infty}{\longrightarrow} \boldsymbol{x}^\infty \ , \quad \boldsymbol{y}^{(k)} \underset{k\to\infty}{\longrightarrow} \boldsymbol{y}^\infty; \tag{7a}$$

$$\|\boldsymbol{x}^\infty - \boldsymbol{y}^\infty\|_2 = \min_{\boldsymbol{x}\in\mathcal{X}, \boldsymbol{y}\in\mathcal{Y}} \|\boldsymbol{x} - \boldsymbol{y}\|_2 \ . \tag{7b}$$

In particular, if $\mathcal{X} \cap \mathcal{Y} \neq \emptyset$*, then* $(\boldsymbol{x}^{(k)})_k$ *and* $(\boldsymbol{y}^{(k)})_k$ *converge to a same point* $\boldsymbol{x}^\infty \in \mathcal{X} \cap \mathcal{Y}$*.*

If disaggregation is not feasible, Thm. 2 states that APM will "converge" to an orbit $(\boldsymbol{x}^\infty, \boldsymbol{y}^\infty)$ of period 2.

The convergence rate of APM has been the subject of several works [18, 20], and it strongly depends on the structure of the sets on which the projections are done: for instance, if the sets are polyhedral, [20, Prop. 4.2] shows that the convergence is geometric. However, there are very few cases in which an explicit upper bound on the convergence rate has been proved. In our case, we are able to obtain such a bound, as shown in the following theorem:

Theorem 3. *For the sets* $\mathcal{X}$ *and* $\mathcal{Y}$ *defined in (3-4), the two subsequences of alternate projections converge at a geometric rate to* $\boldsymbol{x}^\infty \in \mathcal{X}$*,* $\boldsymbol{y}^\infty \in \mathcal{Y}$*, with:*

$$\|\boldsymbol{x}^{(k)} - \boldsymbol{x}^\infty\|_2 \leq 2\|\boldsymbol{x}^{(0)} - \boldsymbol{x}^\infty\|_2 \times \rho_{NT}^k$$

$$\text{where } \rho_{NT} \overset{\text{def}}{=} 1 - \tfrac{1}{4}\left(N(T+1)^2(T-1)\right) < 1 \ ,$$

Same inequalities hold for the convergence of $\boldsymbol{y}^{(k)}$ *to* $\boldsymbol{y}^\infty$*.*

Proof. Appendix II provides a sketch of the proof.

Thm. 3 shows that the APM is efficient in our case of bounded transport polytopes. It shows that the number of iterations for a given accuracy grows linearly in the number of agents N.

As stated in (4), the set $\mathcal{X}$ is a Cartesian product $\prod_n \mathcal{X}_n$, so that the projection $P_{\mathcal{X}}(.)$ can be computed by N projections on $(\mathcal{X}_n)_n$, which can be executed in parallel. Now, instead of solving the quadratic program by standard interior point methods and due to its particular structure, we can use

the algorithm of Brucker [25], which has a complexity in $\mathcal{O}(T)$. On the other hand, $P_{\mathcal{Y}}(.)$ is a projection on an affine space, and the solution can be obtained explicitly as:

$$\forall n,t, \boldsymbol{y}_{n,t} = \boldsymbol{x}_{n,t} + \nu_t \text{ and } \boldsymbol{\nu} = \tfrac{1}{N}(\boldsymbol{p} - \boldsymbol{x}^\top \mathbb{1}_N) \ . \quad (8)$$

B. Generation of a cut from APM iterates

Our key result is the following: in the case where APM converges to a periodic orbit $(\boldsymbol{x}^\infty, \boldsymbol{y}^\infty)$ with $\boldsymbol{x}^\infty \neq \boldsymbol{y}^\infty$ (see Thm. 2), we obtain from $(\boldsymbol{x}^\infty, \boldsymbol{y}^\infty)$ an inequality (6) that is violated by $\boldsymbol{p}$:

Theorem 4. *For the sets $\mathcal{X}$ and $\mathcal{Y}$ defined in (3-4) and if $\mathcal{X} \cap \mathcal{Y} = \emptyset$, the following sets given by the limit orbit $(\boldsymbol{x}^\infty, \boldsymbol{y}^\infty)$ defined in Thm. 2:*

$$\mathcal{T}_0 \stackrel{\text{def}}{=} \{t | p_t > \textstyle\sum_{n\in\mathcal{N}} x^\infty_{n,t}\} \quad (9a)$$

$$\mathcal{N}_0 \stackrel{\text{def}}{=} \{n \ | E_n - \textstyle\sum_{t\notin\mathcal{T}_0} \underline{x}_{n,t} - \sum_{t\in\mathcal{T}_0} \overline{x}_{n,t} < 0\} \quad (9b)$$

define a Hoffman cut of form (6) *violated by $\boldsymbol{p}$, that is:*

$$\sum_{n\in\mathcal{N}_0} E_n - \sum_{t\in\mathcal{T}_0} p_t + \sum_{t\in\mathcal{T}_0, n\notin\mathcal{N}_0} \overline{x}_{n,t} - \sum_{t\notin\mathcal{T}_0, n\in\mathcal{N}_0} \underline{x}_{n,t} < 0 \ . \quad (10)$$

This cut can be reformulated in terms of $\mathbb{1}_N^\top \boldsymbol{x}^\infty$ as:

$$\mathrm{A}_{\mathcal{T}_0} < \sum_{t\in\mathcal{T}_0} p_t \text{ with } \mathrm{A}_{\mathcal{T}_0} \stackrel{\text{def}}{=} \sum_{t\in\mathcal{T}_0}\sum_{n\in\mathcal{N}} x^\infty_{n,t}. \quad (11)$$

Proof. Appendix I gives the sketch of the proof of Thm. 4. The complete proofs will be given elsewhere.

Remark 1. More sophisticated projections methods such as Dykstra's APM [26], EPPM [27], or any method returning outputs $\boldsymbol{x}^\infty$, $\boldsymbol{y}^\infty$ satisfying the same conditions as APM given in Thm. 2, could be used here instead of Algo. 1.

One can see that, intuitively, $\mathcal{N}_0$ is the subset associated to $\mathcal{T}_0$ that minimizes the right hand side of (6). Note that Thm. 4 gives an alternative constructive proof of Hoffman circulation's theorem (Thm. 1) in the case of a bipartite graph of the form of Fig. 1. Moreover, in the case where the disaggregation problem (4) is not feasible, the negation of equation (11) provides a new valid constraint as a condition for the existence of a disaggregated profile of $\boldsymbol{p}$. This constraint can be used in the master problem (II) to update the vector of resources $\boldsymbol{p}$ for the next iteration. This constraint only involves the *aggregate* information $\mathbb{1}_N^\top \boldsymbol{x}^\infty$ on the users profile. To make the process fully *non-intrusive*, we explain in Sec. IV-A how the operator can compute this constraint without making the agents reveal their profiles $(\boldsymbol{x}^\infty_n)_{n\in\mathcal{N}}$.

IV. Non-intrusive Projections and Computation of Disaggregated Optimal Resources

A. Non-Intrusive Alternate Projections Method (NI-APM)

Because of the particular structure of the problem, the projections in APM can be computed separately by the operator and the agents. The projection $P_{\mathcal{Y}}$ is made by the operator, which only requires to know $\boldsymbol{p}$ and the aggregate profile $\boldsymbol{x}^\top \mathbb{1}_N$ according to (8). The projection $P_{\mathcal{X}}$ on $\mathcal{X} = \prod_n \mathcal{X}_n$ is executed in parallel by each agent: n computes $P_{\mathcal{X}_n}$ which only needs her private information E_n and $\overline{\boldsymbol{x}}_n, \underline{\boldsymbol{x}}_n$. However, in the way APM is described in Algo. 1, the operator and the agents still need to exchange the iterates $\boldsymbol{x}^{(k)}, \boldsymbol{y}^{(k)}$ at each step. To avoid the transmission of agents' profiles to the operator, we use a secure multiparty computation (SMC) technique (see [28]) which enables the operator to obtain the aggregate profile $\boldsymbol{S}^{(k)} := \mathbb{1}_N^\top \boldsymbol{x}^{(k)}$ in a non-intrusive manner, as described in Algo. 2.

The main idea of SMC is that, instead of sending her profile $\boldsymbol{x}_n$, agent n *splits* $x_{n,t}$ for each t into N random parts $(s_{n,t,m})_m$, according to an uniform distribution and summing to $\boldsymbol{x}_{n,t}$ (Lines 2-3). Thus, each part $s_{n,t,m}$ taken individually does not reveal any information on $\boldsymbol{x}_n$ nor on $\mathcal{X}_n$, and can be sent to agent m. Once all exchanges of parts are completed (Line 5), and n has herself received the parts from other agents, agent n computes a new aggregate quantity $\boldsymbol{\sigma}_n$ (Line 7), which does not contain either any information about any of the agents, and sends it to the operator (Line 8). The operator can finally compute the quantity $\boldsymbol{S} = \boldsymbol{x}^\top \mathbb{1}_N = \boldsymbol{\sigma}^\top \mathbb{1}_N$.

Algorithm 2 SMC of Aggregate (SMCA) $\sum_{n\in\mathcal{N}} \boldsymbol{x}_n$

Require: Each agent has a profile $(\boldsymbol{x}_n)_{n\in\mathcal{N}}$
1: **for** each agent $n \in \mathcal{N}$ **do**
2: Draw $\forall t, (s_{n,t,m})_{m=1}^{N-1} \in \mathcal{U}([0,A]^{N-1})$
3: and set $\forall t, s_{n,t,N} \stackrel{\text{def}}{=} x_{n,t} - \sum_{m=1}^{N-1} s_{n,t,m}$
4: Send $(s_{n,t,m})_{t\in\mathcal{T}}$ to agent $m \in \mathcal{N}$
5: **done**
6: **for** each agent $n \in \mathcal{N}$ **do**
7: Compute $\forall t, \sigma_{n,t} = \sum_{m\in\mathcal{N}} s_{m,t,n}$
8: Send $(\sigma_{n,t})_{t\in\mathcal{T}}$ to operator
9: **done**
10: Operator computes $\boldsymbol{S} = \sum_{n\in\mathcal{N}} \boldsymbol{\sigma}_n$

Remark 2. As $\boldsymbol{\sigma}_n$, and $\boldsymbol{s}_n$ are random by construction, an eavesdropper aiming to learn the profile $\boldsymbol{x}_n$ of n has no choice but to intercept all the communications of n to all other agents (to learn $(s_{n,t,m})_{m\neq n}$ and $(s_{m,t,n})_{m\neq n}$) and to the operator (to learn σ_n). To increase the confidentiality of the procedure, one could use any encryption scheme (such as RSA [29]) for all communications involved in Algo. 2.

We can use this non-intrusive computation of aggregate $\boldsymbol{S}$ in APM to obtain a *non-intrusive* algorithm *NI-APM* (Algo. 3) in which agents do not reveal neither their profiles nor their constraints to the operator.

One can see that $\boldsymbol{x}$ and $\boldsymbol{y}$ computed in Lines 3 and 8 in Algo. 3 correspond to the projections computed in the original APM Algo. 1. In Algo. 3, the operator obtains the aggregate profile $\boldsymbol{S}^{(k)}$ (Line 5), computes and sends the corrections $\boldsymbol{\nu}^{(k)}$ to all agents (Line 6). Then, each agent can compute locally the projection $\boldsymbol{y}_n^{(k)} = P_{\mathcal{Y}}(\boldsymbol{x}_n^{(k)})$ by applying the correction $\boldsymbol{\nu}^{(k)}$ (Line 8).

Using (8), we get $\boldsymbol{\nu}^{(k)} \to \boldsymbol{\nu}^\infty \stackrel{\text{def}}{=} \frac{1}{N}(\boldsymbol{p} - \mathbb{1}_N^\top \boldsymbol{x}^\infty)$. Thm. 4 uses this limit value through $\mathcal{T}_0{}^\infty \stackrel{\text{def}}{=} \{t \in \mathcal{T} | 0 < \nu_t^\infty\}$. Yet, from APM, one can only access to $\boldsymbol{\nu}^{(k)}$ and thus to

Algorithm 3 Non-intrusive APM (NI-APM)

Require: Start with $\boldsymbol{y}^{(0)}$, $k{=}0$, $\varepsilon_{\text{cvg}}, \varepsilon_{\text{dis}}$, norm $\|.\|$ on $\mathbb{R}^{NT}$

1: **repeat**
2: **for** each agent $n \in \mathcal{N}$ **do**
3: $\boldsymbol{x}_n^{(k)} \leftarrow P_{\mathcal{X}_n}(\boldsymbol{y}_n^{(k-1)})$
4: **done**
5: Operator obtains $\boldsymbol{S}^{(k)} \leftarrow$SMCA($\boldsymbol{x}^{(k)}$) (*cf* Algo.2)
6: and sends $\boldsymbol{\nu}^{(k)} := \frac{1}{N}(\boldsymbol{p} - \boldsymbol{S}^{(k)}) \in \mathbb{R}^T$ to agents $\mathcal{N}$
7: **for** each agent $n \in \mathcal{N}$ **do**
8: Compute $\boldsymbol{y}_n^{(k)} \leftarrow \boldsymbol{x}_n^{(k)} + \boldsymbol{\nu}^{(k)}$
9: **done**
10: $k \leftarrow k+1$
11: **until** $\left\|\boldsymbol{x}^{(k)} - \boldsymbol{x}^{(k-1)}\right\| < \varepsilon_{\text{cvg}}$
12: **if** $\left\|\boldsymbol{x}^{(k)} - \boldsymbol{y}^{(k)}\right\| \leq \varepsilon_{\text{dis}}$ **then**
▷ *found a ε_{dis}-solution of the disaggregation problem*
13: Each agent adopts profile $\boldsymbol{x}_n^{(k)}$
14: **return** DISAG $\leftarrow$ TRUE
15: **else** ▷ *have to find a valid constraint violated by $\boldsymbol{p}$*
16: Operator computes $\mathcal{T}_0 \leftarrow \{t \in \mathcal{T} \mid \frac{3}{2}B\varepsilon_{\text{cvg}} < \nu_t^{(k)}\}$
17: Operator computes $\mathrm{A}_{\mathcal{T}_0} \leftarrow$SMCA($(\boldsymbol{x}_t^{(k)})_{t\in\mathcal{T}_0}$)
18: **if** $\mathrm{A}_{\mathcal{T}_0} - \sum_{t\in\mathcal{T}_0} \boldsymbol{p}_t < 0$ **then**
19: **return** DISAG $\leftarrow$ FALSE, $\mathrm{A}_{\mathcal{T}_0}$
20: **else** ▷ *need to run APM with higher precision*
21: Return to Line 1 with $\varepsilon_{\text{cvg}} \leftarrow \varepsilon_{\text{cvg}}/2$
22: **end**
23: **end**

the *approximation* $\mathcal{T}_0$, computed on Line 16), where B is a pre-defined constant. However, we show that for ε_{cvg} small enough and a well-chosen value of B, we obtain $\mathcal{T}_0 = \mathcal{T}_0^\infty$, so that we get the termination result:

Proposition 1. *For $B > (1-\rho_{NT})^{-1}$, Algo. 3 terminates in finite time.*

The termination of the loop Lines 1-11 is ensured by Thm. 3. In the case where $\left\|\boldsymbol{x}^{(k)} - \boldsymbol{y}^{(k)}\right\| \leq \varepsilon_{\text{dis}}$, Algo. 3 terminates. Otherwise, if $\|\boldsymbol{x}^\infty - \boldsymbol{y}^\infty\| > \varepsilon_{\text{dis}}$, then Algo. 3 terminates (i.e. Line 18 is True and a new cut is found) as soon as $B\varepsilon_{\text{cvg}} < \min\left\{\frac{\|\boldsymbol{x}^\infty - \boldsymbol{y}^\infty\|_1}{2\sqrt{N}}, \frac{2}{5}\underline{\nu}\right\}$, where $\underline{\nu} \stackrel{\text{def}}{=} \min\{|\nu_t^\infty| > 0\}$ and with $\|.\| = \|.\|_2$. The complete proof is omitted here.

In practice, we can start with a large ε_{cvg} to obtain the first constraints while avoiding useless computation, and then half ε_{cvg} if needed (Line 21) until the termination condition holds.

Remark 3. Lagrangian decomposition is another promising technique to develop privacy-preserving algorithms. However, Lagrangian decomposition requires convexity assumptions, whereas in the present approach, combining polyhedral cuts and alternate projection methods, the optimization problem can be nonconvex (we shall actually solve such a nonconvex example in Sec. V).

B. Non-intrusive Disaggregation of Optimal Allocation

In this section, we describe a method to compute a solution of the global problem (1), that is, an optimal resource allocation $\boldsymbol{p}$ for which a disaggregation exists, along with an associated disaggregated profile $\boldsymbol{x}_n$ for each agent n. This computation is done in a non-intrusive manner: the operator in charge of $\boldsymbol{p}$ does not have access neither to the bounding constraints $\overline{\boldsymbol{x}}$ and $\underline{\boldsymbol{x}}$ of the agents nor to the agents disaggregated profile $\boldsymbol{x}$, as detailed in Algo. 4 below.

Algorithm 4 Non-intrusive Optimal Disaggregation

Require: $s = 0$, $\mathcal{P}^{(0)} = \mathcal{P}$; DISAG = FALSE

1: **while** Not DISAG **do**
2: Compute $\boldsymbol{p}^{(s)} = \arg\min_{\boldsymbol{p}\in\mathcal{P}_{\text{cs}}^{(s)}} f(\boldsymbol{p})$
3: DISAG $\leftarrow$ NI-APM($\boldsymbol{p}^{(s)}$)
4: **if** DISAG **then**
5: Operator adopts $\boldsymbol{p}^{(s)}$
6: **else**
7: Obtain $\mathcal{T}_0^{(s)}, \mathrm{A}_{\mathcal{T}_0}^{(s)}$ from NI-APM($\boldsymbol{p}^{(s)}$)
8: $\mathcal{P}^{(s+1)} \leftarrow \mathcal{P}^{(s)} \cap \{\boldsymbol{p} | \sum_{t\in\mathcal{T}_0^{(s)}} p_t \leq \mathrm{A}_{\mathcal{T}_0}^{(s)}\}$
9: **end**
10: $s \leftarrow s+1$
11: **done**

Algo. 4 iteratively calls NI-APM (Algo. 3) and in case disaggregation is not possible (Line 6), a new constraint is added (Line 8), obtained from the quantity $\mathrm{A}_{\mathcal{T}_0}$ defined in (11), to the resource problem (II). This constraint is an inequality on $\boldsymbol{p}$ and thus does not reveal significant individual information to the operator. The algorithm stops when disaggregation is possible (Line 4). The termination of Algo. 4 is ensured by the following property and the form of the constraints added (10):

Proposition 2. *Algo. 4 stops after a finite number of iterations, as at most 2^T constraints (Line 8) can be added to the master problem (Line 2).*

Although there exist some instances with an exponential number of independent constraints, this does not jeopardize the proposed method: in practice, the algorithm stops after a very small number of constraints added (see the example of Sec. V). Intuitively, we will only add constraints "supporting" the optimal allocation $\boldsymbol{p}$.

Remark 4. Algo. 4 solves problem (1) in a privacy-preserving manner for agents. For this, we use both the results of Thm. 4 and SMC to *securely* transmit the aggregate profile to the operator. For the latter point, other techniques could be used such as the consensus-based aggregation algorithm in [30]. A comparison of the different possible techniques, relying on quantitative privacy indicators, would be interesting, and is an avenue for further work.

V. APPLICATION TO MANAGEMENT OF A MICROGRID

We apply the proposed method to solve a nonconvex distributed problem in the energy field. We consider a microgrid [31] composed of N electricity consumers with flexible appliances (such as electric vehicles or water heaters), a photovoltaic (PV) power plant and a conventional generator.